sheet (8)

Computer Architecture

Sheet (8)

5.1 Give a block diagram similar to the one in Figure 5.10 for a 8M $\times$ 32 memory using 512K $\times$ 8 memory chips.

5.4 Consider a main memory constructed with SDRAM chips that have timing requirements depicted in Figure 5.9, except that the burst length is 8. Assume that 32 bits of data are transferred in parallel. If a 133-MHz clock is used, how much time does it take to transfer:

(*a*) 32 bytes of data

(*b*) 64 bytes of data

What is the latency in each case?

5.5 Criticize the following statement: "Using a faster processor chip results in a corresponding increase in performance of a computer even if the main memory speed remains the same."

5.9 A block-set-associative cache consists of a total of 64 blocks divided into 4-block sets. The main memory contains 4096 blocks, each consisting of 128 words.

(*a*) How many bits are there in a main memory address?

(*b*) How many bits are there in each of the TAG, SET, and WORD fields?

5.10 A computer system has a main memory consisting of 1M 16-bit words. It also has a 4K-word cache organized in the block-set-associative manner, with 4 blocks per set and 64 words per block.

(*a*) Calculate the number of bits in each of the TAG, SET, and WORD fields of the main memory address format.

(*b*) Assume that the cache is initially empty. Suppose that the processor fetches 4352 words from locations 0, 1, 2, . . . , 4351, in that order. It then repeats this fetch sequence nine more times. If the cache is 10 times faster than the main memory, estimate the improvement factor resulting from the use of the cache. Assume that the LRU algorithm is used for block replacement.

5.15 How might the value of k in the interleaved memory system of Figure 5.25*b* influence block size in the design of a cache memory to be used with the system?

5.16 In many computers the cache block size is in the range of 32 to 128 bytes. What would be the main advantages and disadvantages of making the size of cache blocks larger or smaller?

5.17 Consider the effectiveness of interleaving with respect to the size of cache blocks. Using calculations similar to those in Section 5.6.2, estimate the performance improvement for block sizes of 16, 8, and 4 words. Assume that all words loaded into the cache are accessed by the processor at least once.

تكمله السؤال الموجود
ف الورق ...

Sheet #9 (Memory)

$\boxed{5.1}$ Small memory : $512K \times 8 = 2^{9} * 2^{10} * 8 = 2^{19} * 8$

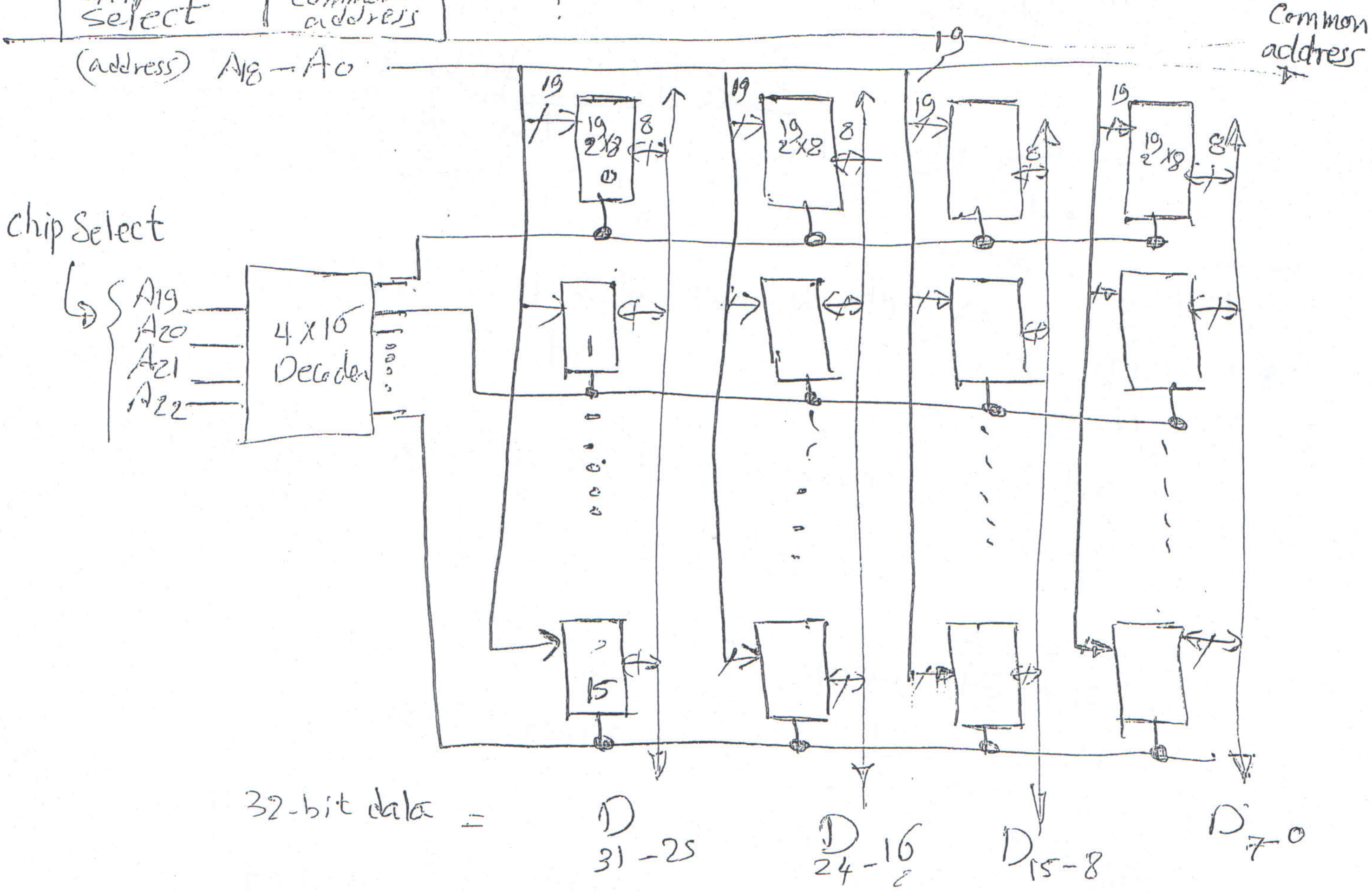
Large memory : $8M \times 32 = 2^{3} * 2^{20} \times 32 = 2^{23} \times 32$

Number of needed Chips $= \dfrac{2^{23} \times 32}{2^{19} \times 8} = 2^{4} * 2^{2} = 16 \times 4$

عمق ← 16   4 → عرض

Small memory	Large memory
$2^{19} \times 8$	$2^{23} \times 32$
Common address lines $= \boxed{19}$	address Lines $= \boxed{23}$
Data Lines $= 8$	Data lines $= 32$
	Chip select (address inputs to Decoder) $=$ $23 - 19 = 4$
	$\therefore$ we need decoder $4 * 16$

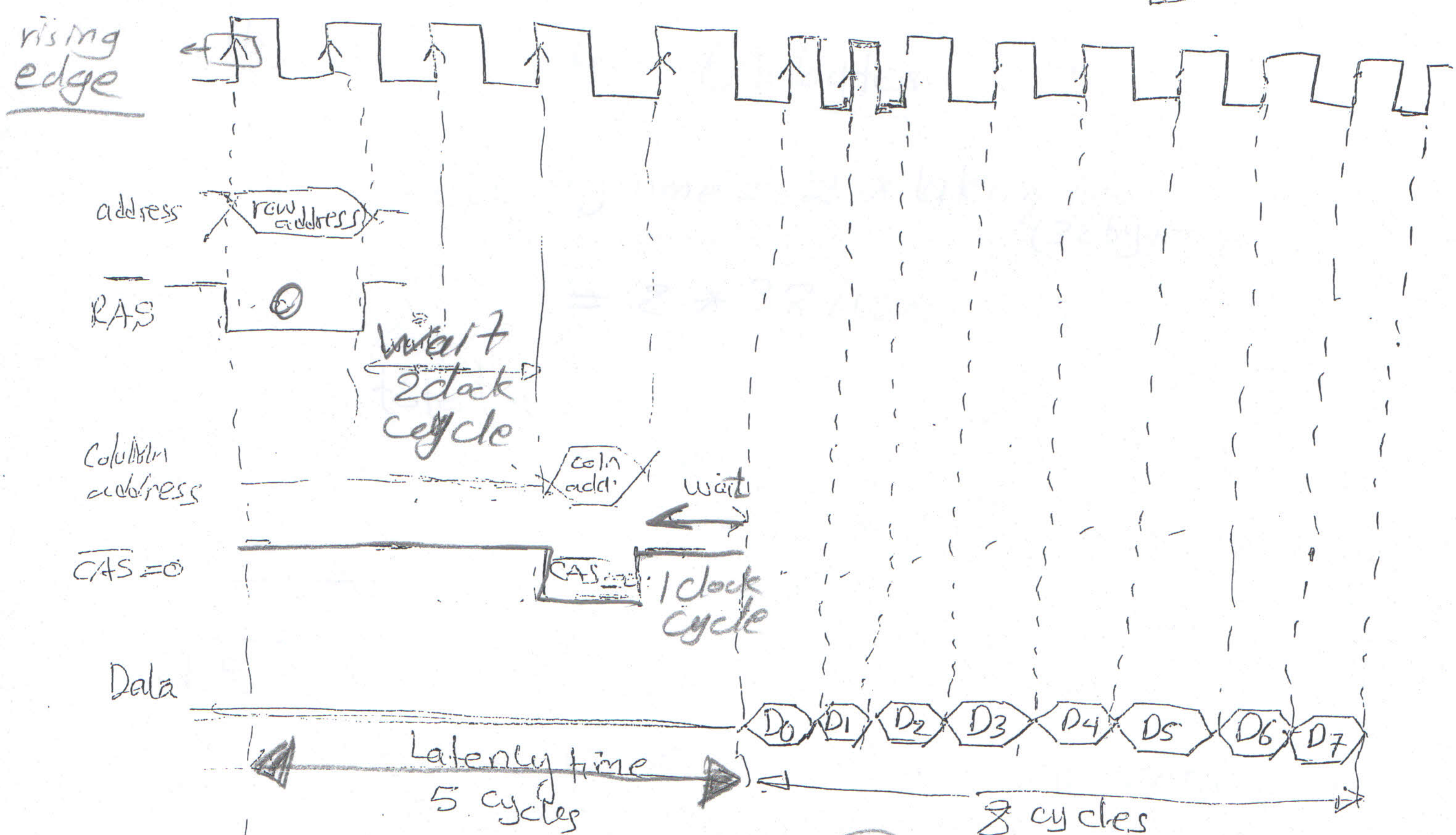
$A_{22} \dots A_{19}$	$A_{18} \dots A_{0}$
chip select	Common address



5.4

synch0 $\xrightarrow{by}$ clock

Burst read of length 8



Burst read of length 8

نستطيع قراءة 8 word وكل word عرضها 32 bit

$\frac{8 \text{ word}}{32 \text{ bit}}$

∴ total Read data (Band width) (BW) $= 8 \text{ (عدد words)} * 32 \text{ bit (عرض word)} / 8$

$= 8 * 4 \text{ bytes}$

$= 32 \text{ bytes}$

(a) When we need to transfer 32 byte

✓ latency time = 5 clock cycles $* T_i$

$$T_i = \frac{1}{f} = \frac{1}{133\,MHz} = \frac{1}{133 * 10^6} =$$

∴ latency time $= 5 * \frac{1}{133 \times 10^6} = 28\ \mu sec$

✓ total time need to transfer 32 byte $= 13 * T_i = 13 * \frac{1}{133 \times 10^6} = 98\ ns$

(b) <u>to transfer 64 bytes</u>

$$\text{latency time} = 2 * \text{latency time}'_{(32\,bytes)}$$

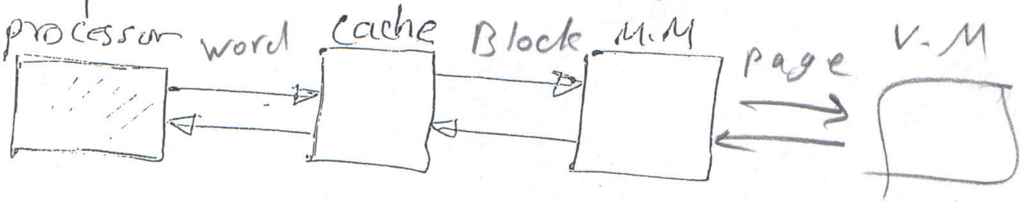
$$= 2 * 38\ \text{nsec}$$

$$\text{total time}_{64\,byte} = 2 * \text{total time}'_{32\,byte}$$

$$= 2 * 98\ \text{nsec} = 196\ \text{nsec}$$

$\boxed{5.5}$

A faster processor chip will result in increasing performance, but the amount of increase will not be directly proportional to increase in processor speed, because Cache miss penalty will remain the same if the main memory speed is not improved.



استخدام معالج بسرعة عاليه لاجل انه يزيد من الكفاءة
ولكن كمية الزيادة هذه ليست بالضرورة أن تتناسب مع
سرعة المعالج المضافة وذلك بسبب أن وجود العديد
من Cache Miss يؤثر على الكفاءة ∴
∴ لابد من رفع سرعة (main memory) هي الاخرى حتى نقلل من
<u>(Cache Miss)</u>

$\boxed{5.9}$

Cache	Main memory
# of cache blocks $= 64 = 2^6$	# of block in M.M $= 4096 = 2^{12}$
$2^2 = 4$ blocks $\rightarrow$ 1 set	
2^6 block $\rightarrow$? set	1 block $\rightarrow$ 128 word $= 2^7$
# of sets in cache $= \frac{2^6}{2^2} = 2^4$ $\boxed{= 16}$	

main memory address

Tag	set	word
8	4	7

= 19 bit address

1 block $\rightarrow 2^{7}$ word

# of sets in cache 2^{4}

$\frac{\text{\# of block M.M}}{\text{\# of in cache}} = \frac{2^{12}}{2^4} = 2^{8}$

(5.10) →

(a)

Tag	Block	word
6	6	7

→ 19

2^7 word

chache of block

(b) $\frac{2^{12}}{2^6} = 2^6$

Tag	word
12	7

→ 19

2^7 word

block in M.M $= 2^{12}$

5.10

MM

1 M → 16 bit word

$\boxed{4 \text{ blocks}}$ ← 1 Set

64 word ← 1 block

# of block in MM

$= \frac{1M}{64} = \frac{2^{20}}{2^6} = 2^{14}$

of

cache

4K word Cache

$2^6 = 64$ word ← 1 block

4K word ← ? block

# of blocks in Cache $= \frac{4K}{64}$

$= \frac{2^2 * 2^{10}}{2^6} = \frac{2^{12}}{2^6} = 2^6 = 64$ K

↳ block

# of sets $= \frac{2^6}{4} = \frac{2^6}{2^2} = 2^4$

4 blocks ← 1 set

2^6 block → ?

1

Tag	Block	word
8	6	6

Block: in chache, # of block in cache 2^6

→ 1 block → 64 word $= 2^6$

$\frac{B_{MM}}{B_{cache}} = \frac{2^{14}}{2^6} = 2^8$

2

Tag	word
14	6

3

Tag	Set	word
10	4	6

Set: in chache

Sets

$\frac{B_{MM}}{} = \frac{2^{14}}{24} = 2^{10}$

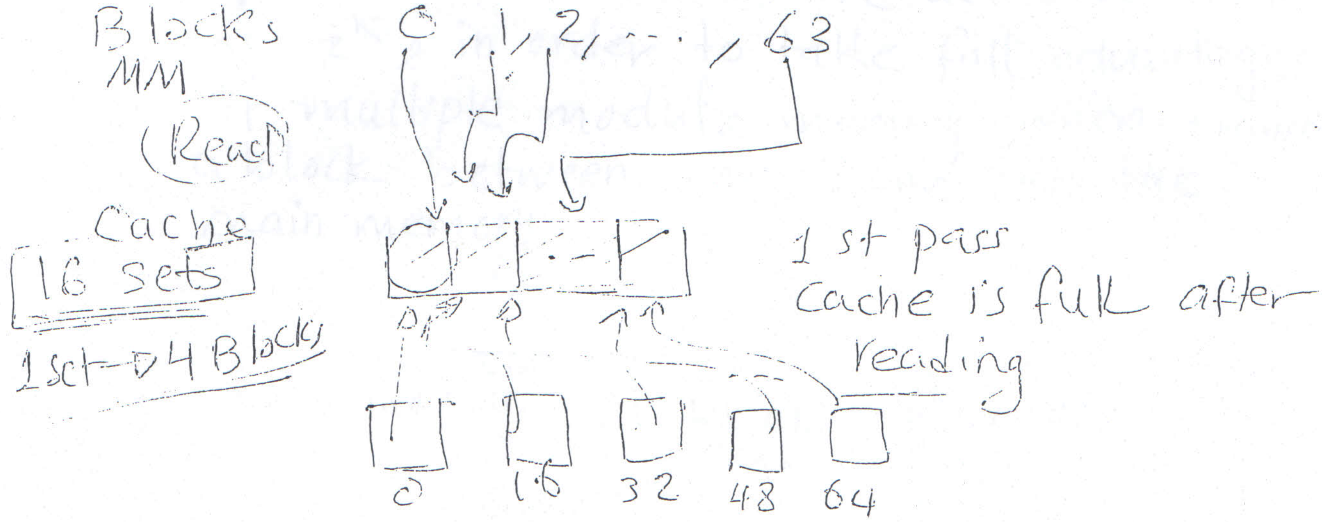
(b)

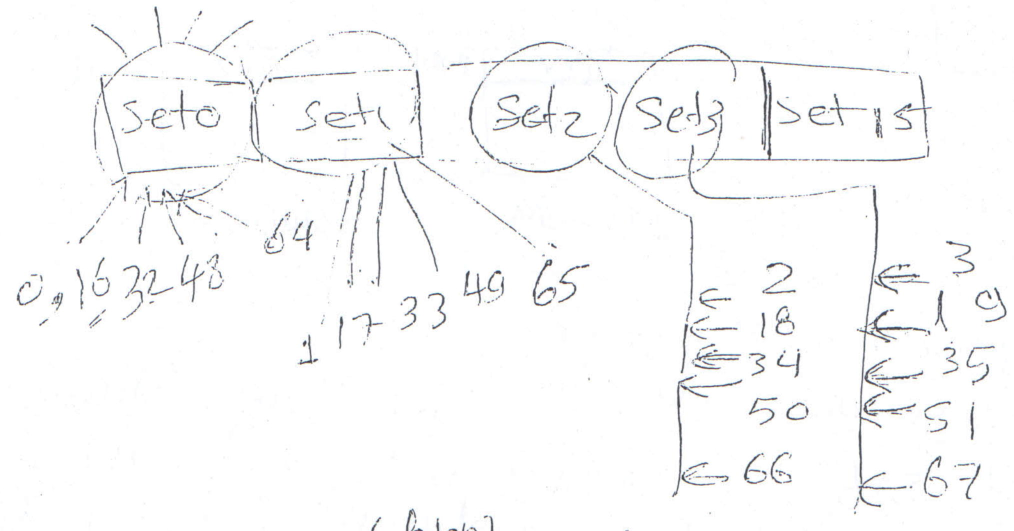
Improvement factor = Time without Cache / Time with Cache = $\dfrac{\overset{\text{passes}}{10} * 68 * 10\tau}{(1 * 68 * 11\tau) + 9(20 * 11\tau + 48 \times 1\tau)}$ → for each Block

pass الأول — pass [illegible] — Cache — Block [illegible] Cache

Words 0, 1, 2, ----, 4351

occupy Blocks 0 —— to 67 in MM (68 block)





ال (Blocks) [illegible] (fetch) [illegible] Cache [illegible] first pass

(9 Passes)

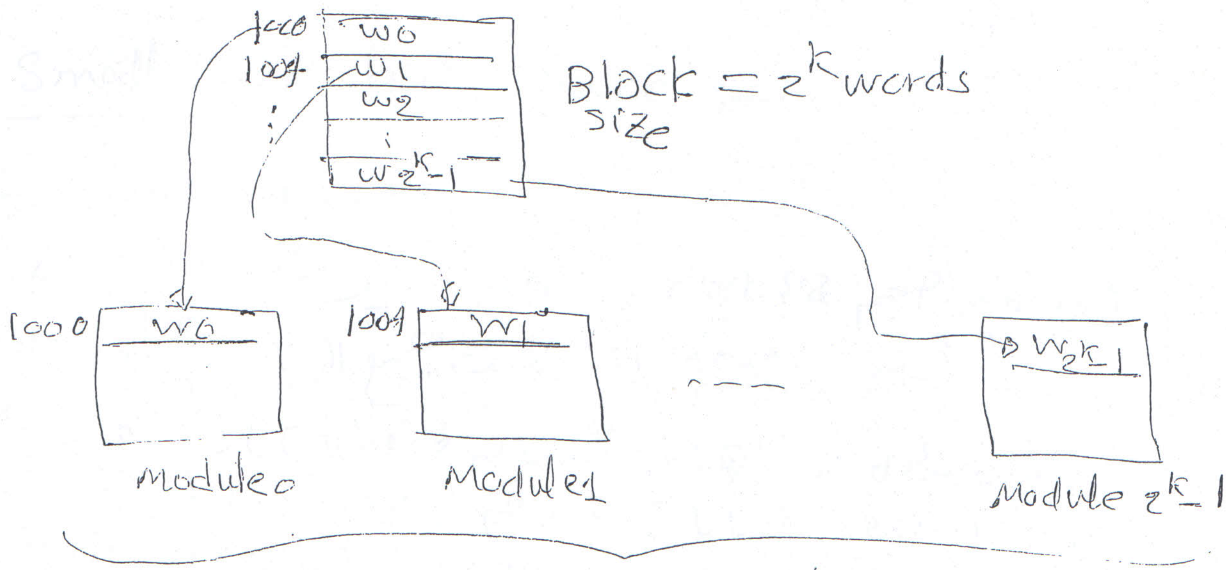
1 st pass 68 Block MM $\xrightarrow{\text{read}}$ Cache

9 scussive passes ($4_{\text{Blocks}} * 12_{\text{sets}} = 48$) ⟶ are found in Cache for all passes

Remaining Blocks (68 − 48) = 20 → must be fetched from MM

$\boxed{5.15}$

The block size (number of words in a block) of the cache should be at least as large as 2^k, in order to take full advantages of multiple module memory when transfering a block between the cache and the main memory



Memory Interleaving

توزيع (words) الخاصة بال Block على جميع Modules
فإذا قمنا بجعل Block مساوي لعدد من words
مساوي لعدد ال (Modules) المتاحة في الذاكرة
فإننا نستطيع قراءة جميع words لهذا ال (Block)
من الذاكرة إلى Cache في نفس الوقت .

$\boxed{5.16}$ ⇒

① Large size cache Block advantages :

ⓐ fewer misses if most of the data in the block are actually used.

Disadvantages :

wasteful if much of the data are not used before the cache block is ejected from the cache.

■ Small size block cache :-

– more misses

إذا كان حجم Block size في Cache كبير يؤدي ذلك إلى

fewer misses إذا كانت جميع البيانات المطلوبة موجودة في Block

العيوب يؤدي إلى إهدار المساحة في (Cache) إذا كانت جميع

أو بعض البيانات التي تم إحضارها لن نحتاجها

حتى يرجع (Block) من (Cache) إلى الذاكرة.

Time without cache = 4.04 * 25 * Ti

= 101 Ti ← cycle time

(1.17) $\boxed{5.17}$

for black size 16 words:-

with interleaving ⟶ $M = 1 + 8 + 4 + 4 + 4 + 4 = 21$ cycles

adders, MM → DBR, DBR → cache, DBR i → cache i

Time without

effectivness = $\frac{\text{Time without cache}}{\text{time with cache}} = 4.04$

total time without Int $= 1 + 8 + (7 * 4) + 1 = (25) \times Ti$

8 word Block:-

$M = 1 + 8 + 4 + 4 = 17$ cycles

to brought two 8-word Block

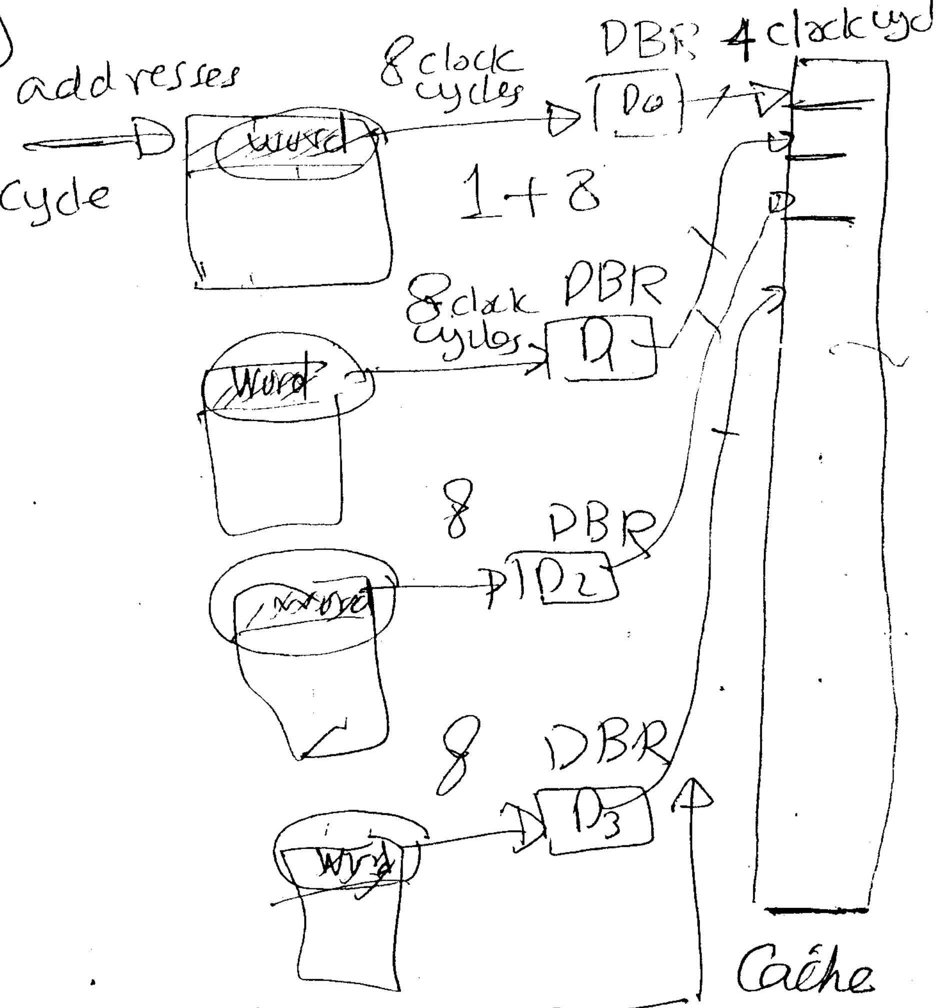
$M = 2 * 17 = 34$ cycles

effectivness = $\frac{\text{Time without cache}}{\text{Time with cache}} = 3.3$

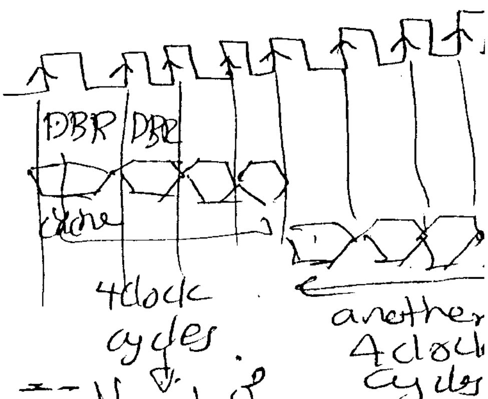
↳ 2 words:-

total time with $= 1 + 8 + 4 + 4 + 4 + 4 + 4 + 4 + 4 + 4$

Interleaving



يتم نقل البيانات الموجودة في Modules من DBR at the same time
في الذاكرة إلى الـ (Cache) واحد يلو الآخر عن
طريق في (4 clock cycles)



في نفس الوقت يتم تحميل (4 words) متتاليه في DBR

وبعد 4 clock cycles فإنها تنتقل. 4 clock cycles إضافية

لإرسال words الجديدة
(2) DBR إلى (Cache) ← word تلو الأخرى.

total time needed = 1 + 8 + 4